

REMARKS/ARGUMENTS

Prior to the entry of this Amendment, claims 1-24 were pending in this application. No claims are amended, no claims are added, and no claims are canceled herein. Therefore, claims 1-24 remain pending. Applicants respectfully request reconsideration of these claims for at least the reasons presented below.

Replacement Specification

As requested by the Examiner, a Replacement Specification is re-submitted herewith. Please find attached a clean version (Appendix II, Replacement Specification - Clean Copy) and a marked-up version (Appendix I, Replacement Specification - Marked-up Copy) showing revisions. Revisions to the specification have been made to remove shading from the tables included therein, to correct typographical and grammatical errors, and to correct other formal matters as summarized above. The Applicants, therefore, respectfully submit that the replacement specification contains no new matter and request entry of this specification.

Title

As noted above, and in the previous Amendment, the Applicants request the title to be changed to "Responding To Requests For Remote Operation In Systems With Multiple Servers." No indication has yet been given whether this title is acceptable. The Applicants respectfully request that the title be changed as requested or, if this title is not acceptable, the Applicants be so informed and given a chance to submit a revised title.

35 U.S.C. § 102 Rejection, Freeman

Claims 1-24 have been rejected under 35 U.S.C. § 102(e) as being anticipated by U. S. Patent No. 6,785,726 to Freeman et al. (hereinafter "Freeman"). The Applicants respectfully submit the following arguments pointing out significant differences between claims 1-24 submitted by the Applicants and Freeman.

“A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.” MPEP 2131 citing *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Applicants respectfully argue that Freeman fails to disclose each and every claimed element. For example, Freeman fails to disclose, either expressly or inherently, causing the destination system to block new requests. Rather, Freeman teaches either starting a new thread to concurrently handle events in a multi-threaded system or queuing events to be handled, presumably, in the order received in a single-threaded system. In either case, Freeman teaches that new requests will be received by the destination system while previous requests are being processed.

As noted previously, Freeman “relates to a method and apparatus for delivering events to local and remote servers.” (Column 1, lines 45-46.) Freeman teaches “a first server having a first plurality of subsystems and a first event bus and a second server having a second plurality of subsystems and a second event bus.” (Column 1, lines 49-51.) The method of Freeman includes “posting to the first event bus, by one of the first plurality of subsystems, a local event having as its target another of the first plurality of subsystems [and] . . . posting to the first event bus, by one of the first plurality of subsystems, a remote event having as its target one of the second plurality of subsystems.” (Column 1, lines 52-57.) These events include request events “that send a request for service or functionality to another subsystem on the same server or to a remote server in the server farm” and reply events that “occur in response to request events.” (Column 27, lines 56-65.) Freeman also discloses a “SendEventAndWait” command for issuing a request event. (Column 30, line 60, through column 32, line 36.)

Under Freeman, a source subsystem can issue a request using the “SendEventAndWait” command and then “block further execution of the thread that generated the request event until the response from the destination subsystem is received.” (Column 30, line 65, through column 31, line 7.) After the request is issued, a target can be located from a service locator (column 31, lines 10-24) and the request can be sent to the target/destination

system via the transport layer and event bus (column 31, lines 25-43). The way in which the destination system handles the event depends on whether the destination system is single-threaded or multi-threaded. (Column 31, lines 41-43.) If the destination subsystem is multi-threaded, the destination system executes an appropriate handler routine for responding to the request event. (Column 31, lines 44-50.) If the destination subsystem is single-threaded, the destination system places a pointer to the event buffer holding the request event in the event queue associated with the destination subsystem. (Column 31, lines 51-59.)

That is, upon receiving an event, the destination system of Freeman either handles the event immediately or places the event in a queue for later handling depending on whether the system is multi-threaded or single-threaded. However, Freeman does not disclose causing the destination system to block new requests. To the contrary, Freeman teaches either starting a new thread to concurrently handle events in a multi-threaded system or queuing events to be handled, presumably, in the order received in a single-threaded system. In either case, Freeman teaches that new requests will be received while previous requests are being processed. Therefore, Freeman does not teach the destination system blocking new requests.

In response to these arguments, the Office Action agrees that “Freeman does not disclose causing the destination system to block new requests.” (Office Action, page 4, ¶ 16, sub. a, first sentence.) The Applicants fully agree with this statement and, based on this apparently mutual understanding of Freeman, believe the claims to be in condition for allowance.

However, the Office Action then goes on to argue that “since the event was queued and processing not begun/started, such was/is the equivalent of blocking (i.e., the request was/is blocked from the run process/state) the request from the run state and from the actual processor and associated software running thereon.” The Applicants respectfully but strongly disagree with this reasoning. Freeman’s teaching of placing requests to the destination system into a queue cannot reasonably be seen as equivalent to blocking such requests. To the contrary, Freeman’s disclosure of the destination system placing requests into a queue indicates Freeman’s

intention for the destination system to continue to receive requests even if the destination system is a single-threaded system that is busy handling a previous request.

The Office Action further argues that “since the thread was blocked from further execution, the thread was unable to generate new requests thus also blocking any subsequent request from the thread until the server completed service or requests in progress.” Again, the Applicants respectfully but strongly disagree with this reasoning. If there is no request, that is, it has not yet been made by the source, how can such a non-existent request be blocked by the destination system?

Claim 1, upon which claims 2-10 depend, claim 11, upon which claims 12-17 depend, and claim 18, upon which claims 19-24 depend, each recite in part the “remote server blocking new requests in response to said requests.” Freeman does not disclose the remote server, i.e., destination system, blocking new requests in response to a request. Rather, Freeman teaches either starting a new thread to concurrently handle events in a multi-threaded system or queuing events to be handled, presumably, in the order received in a single-threaded system. In either case, Freeman teaches that new requests will be received by the destination system while previous requests are being processed. For at least these reasons, claims 1-24 are distinguishable from Freeman and should be allowed.

Conclusion

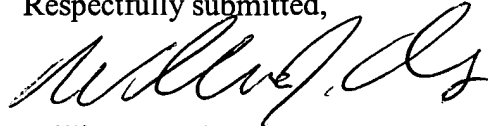
In view of the foregoing, Applicants believe all claims now pending in this application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

Application No. 09/997,408
Amendment dated April 14, 2006
Amendment under 37 CFR 1.116 Expedited Procedure
Examining Group 2142

PATENT

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 303-571-4000.

Respectfully submitted,



William J. Daley
Reg. No. 52,471

Date: April 14, 2006

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, CA 94111-3834
Tel: 303-571-4000
Fax: 415-576-0300
WJD/jln
60712954 v1